Biskra University **10/03/2025**
Computer Science department
Course: Formal Languages (TL)

Problem set 3

Exercise 1:

Build a finite state automata accepting the following language: '**strings containing an even number of a or an odd number of b'.**

Exercise 2:

- Build a finite state automata accepting the following language: "**all the strings of {a,b}* containing the word aab**"
- Build the equivalent deterministic automata.

Exercise 3:

1- Build a non-deterministic automata A recognizing the language L over the alphabet V = {a, b} such that all the words of L simultaneously satisfy the following two conditions:

- Any word of L has a **length divisible by 3**
- Any word of L begins with the symbol **a** and ends with the symbol **b**

2- Build the equivalent deterministic automata.

Exercise 4:

Build a finite state automata accepting the following languages:

1- L= { w Ɛ Vt* / |w|≡0 [p] p> 0 } Vt= {a, b }
2- L= { w Ɛ Vt* / w= a^nb^m , m+n ≡ 0 [p] p> 0 } Vt= {a, b }

Exercise 5: (passage : grammar-FSA- RE)

Find the regular expressions defining the languages generated by the following grammars:

1- S→ bS / aX X→ bS / aY Y → aY / bY / a / b

2- S → aaS / abS / baS / bbS / Ɛ

Exercise 6: (passage : grammar-FSA- RE)

Find a regular grammar equivalent to the following grammar:

1- S→ XYZ X → aX / bX / Ɛ Y → aY / bY / Ɛ Z → aZ / Ɛ

2- S→ XY X → aX / Xa / a Y → aY / Ya / a

Exercise 7: (passage : grammar-FSA- RE)

Find an equivalent grammar that does not contain the empty word (Ɛ)

1- S→ aX / bX X→ a / b / Ɛ
2- S→ aS / bX X → aX / Ɛ

Exercise 8:

Build a deterministic finite state automata accepting the language defined by the following regular expression:

1- (00 + 11)* (01 +10) (00 + 11)*
2- ((a+b)* abb (a+b)*)*

Exercise 9:

Give a regular expression that describes the possible words for a 24-hour digital clock hours where hours, minutes and seconds are separated by ":"

Exercise 10:

Let the FSA defined by the following transition table: **A**: initial state **E :** Final state

	0	1
A	B	C
B	B	D
C	B	C
D	B	E
E	B	C

1- Using Arden theorem, find the regular expression equivalent to this automata
2- Minimizing this automata
3- Using Arden theorem, find the regular expression equivalent to the minimized automata

Exercise 11:

Let the FSA defined by the following transition table: A: initial state C : Final state

	a	b
A	B	F
B	G	C
C	A	C
D	C	G
E	H	F
F	C	A
G	G	E
H	G	C

1- Minimizing this automata
2- Using Arden theorem, find the regular expression equivalent to the minimized automata

Exercise 12: (do it alone)

Let L1 be the language of the words of {a, b}* containing an odd number of letters "a"; and L2 = {aa, ab}.

1) Build a simple finite state automata that accepts L1.
2) Build a simple finite state automata that accepts L2.
3) Build a simple finite state automata that accepts L1 ∪ L2.
4) Make the automata deterministic.
5) Minimizing the deterministic automata.

Exercise 13: (do it alone)

Consider the grammar G, where R = { A → aaB | abC; B → aC | To ; C → bC | ε }

1) Build the deterministic automata that accepts the iteration of L(G).

2) Build the automata that accepts the mirror of L(G).